

ABCS—Not Business as Usual

Brig. Gen. Harold J. Greene, USA ■ Janet Greenberg



In order to fight and win wars, the Army must be organized, directed, controlled, supported, and sustained in a manner that guarantees mission accomplishment. The Army of the 21st century combines flexible organizational elements, battle-proven techniques of leadership, and a basic concept of command and control that combines historical experience and modern information technology. The Army Battle Command System (ABCS) provides commanders the battle command architecture necessary to gain and maintain the initiative and successfully execute missions.

ABCS consists of 11 battlefield systems that provide capabilities to support the warfighter's mission needs. Each system aids in planning, coordinating, and executing operations by providing access and passing information from a horizontally integrated command and control network.

Greene previously served as the U.S. Army Project Manager, Battle Command, where he led the development and fielding of the ABCS system of systems. He is currently the deputy commanding general, U.S. Army Research, Development and Engineering Command/commanding general, Natick Soldier Systems Center. **Greenberg** currently works for Product Manager Strategic Battle Command. She has more than 25 years of software and systems engineering experience in military and commercial applications.

Systems within ABCS support soldiers specializing in warfighting functions; for example, maneuvering, intelligence, fire support, and logistics. In 2003, the chief of staff of the Army directed that the Army shift its funding efforts away from developing the battle command architecture using a bottom-up and functionally aligned structure to one that is focused on developing the architecture from the top down with greater horizontal integration. Additionally, he directed fielding the ABCS capability to the entire Army.

Program Manager Battle Command (PM BC) was designated as the ABCS lead shortly thereafter, responsible for delivering an interoperable version of ABCS to the Army per the chief of staff's guidance. The PM's team met the intermediate milestones of delivering ABCS 6.4 to the Central Technical Support Facility for integration testing in April 2004, training soldiers in the 4th Infantry Division in fall 2004, participating in Joint Red Flag/Roving Sands in summer 2005, and then delivering a final version to the 4th Infantry Division in support of their deployment in fall 2005. Since then, every brigade combat team rotating into theater has received ABCS equipment, training, and support.

Implementation Challenges

There were many lessons learned in those two years during the test, training, and fielding of the software. The greatest challenge that arose was that the software was found to be stovepiped, or functionally aligned. It was obvious to users of ABCS that each of the component systems was designed and developed independently of the others. Each system had unique user interfaces, servers, training products, and field support mechanisms. The resulting system of systems, while more interoperable than its precursor systems, was complicated to train and maintain. In addition, similar capabilities were provided by multiple systems within ABCS.

Additionally, the world did not stand still while ABCS 6.4 was developed, tested, and fielded. The Army continued to evolve, the force structure changed, and modular force packages were built around brigades. The commercial world improved many key technologies such as voice over IP, satellite communications, and Web-enabled software. Finally, future battle command programs loomed on the horizon. How would ABCS be phased out while still meeting the needs of soldiers in conflict? How best to manage the change? To answer those questions, the Battle Command Migration Plan was developed in 2005.

Developing a Vision for the Future

The Battle Command Migration Plan mapped out the development, fielding, and finally, retirement path for ABCS. The goals of the plan were to lower life cycle cost by moving to a smaller footprint; make the systems easier to use, train, and configure; and field a single standard capability to every unit that provides the basis for tailoring for unique unit and mission needs.

The vision presented in the plan had three primary components roughly corresponding to the main thrusts of the work to implement future ABCS planning—technical, logistic, and contracting. Taken together, those three visions formed the programmatic baseline that was implemented in the move from stovepipe systems to a net-centric force. The technical vision's main theme moved the stovepipe systems from a server-centric architecture to a net-centric architecture. The logistics vision focused on breaking the existing stovepipe paradigm. One key work area was to provide a single interface to the field for all ABCS systems to make it easier for soldiers to report issues and track fixes. The contracting vision supported a more agile software development environment. It allowed for smaller contracting houses to develop services, but envisioned a single integration organization performing the systems engineering top-level design tasks and managing the necessary integration and test efforts. Those visions were not independent of each other and thus required close coordination amongst all involved stakeholders.

Need for Greater Coordination

In 2003, each component of ABCS was in its own program shop with a program manager and prime developer. Each program shop developed and tested all component functionality. For example, the fires part of ABCS, Advanced Field Artillery Tactical Data System, had a program office with a program manager with a single prime developer. The program manager, as stated in his charter from the assistant secretary of the Army for acquisition, logistics and technology, was responsible only for delivering the fires capability in a product to the Army. The wording of the charters, with their well-defined scope of responsibility, caused program managers to become very product-focused.

With program managers so product-focused, there was a tendency to neglect system-of-systems considerations when faced with decisions. One example is how Battle Command Sustainment and Support System (BCS3) chose a laptop for fielding in 2003. The original decision was to move from the common hardware platform to a Microsoft® Windows laptop. An analysis was done, but neglected to factor in system-of-systems requirements. An IBM® laptop was selected, based mainly on cost considerations, for the BCS3 program. Unfortunately, the IBM laptop did not become the standard platform for ABCS and thus required a unique maintenance system, confusing the field users. At one point, there were four different methods to sustain laptops provided by various parts of ABCS. While the choice of the IBM was correct for BCS3, it was not best for the system of systems.

This way of analyzing requirements—taking into account the entire battle command suite of systems—required a transformation in the thinking of leadership and developers. The emphasis in PM BC shifted from working on a single product to working on both the product and the system of systems.

Emphasis was also placed on working for both the warfighter and the taxpayer to make sure the capability was developed in the most cost-effective way possible.

Working in a system-of-systems environment required greater coordinated activity across the programs and the functional areas. Take, for example, the server consolidation effort begun in 2004. This effort was spearheaded by Product Manager Tactical Battle Command (PdM TBC). By hosting core server functionality, the number of servers required in theater was greatly reduced. This effort was a great example of team play—PdM TBC needed to work closely with the other ABCS program manager shops so that redundant servers could be identified and then eliminated from the system architecture. The technical and fielding teams had to determine how this new server would be delivered and maintained in theater. Finally, the training team had to work to determine how best to train soldiers and maintainers on this new server.

The design, development, testing, training, and fielding of a system of systems also required a different mindset for Headquarters, Department of the Army and the Office of the Secretary of Defense. Systems and funding have traditionally been set for individual systems, not a system of systems. Budgeting, for example, is done on an individual system basis; however, there are some activities that are required at a system-of-systems level (such as getting certifications and doing system-of-systems systems engineering) that either need to be funded directly or via taxing of the component systems.

Leadership Support

Perhaps the biggest lesson learned in the ABCS process is that leadership must play a key role. Leaders must be forward thinking and create processes that force engagement by the product managers and the other stakeholders. PM BC instituted the Executive Integrated Product Team with sub-IPTs for technical issues, training, and fielding. The EIPT was used as a forcing function to get different disciplines and all the program manager shops to communicate and to synchronize large groups who often had different goals and schedules.

Need for Systems Engineering

For a system of systems to be developed and fielded, there must be upfront systems engineering performed. That includes hardware, software, network architecture descrip-

tions, risk identification and mitigation plans, data and schema descriptions, and integrated schedule development. Such critical planning must also include periodic public design reviews.

The first step in the systems engineering process might include, as it did for PM BC, a briefing with the system-of-systems technical, sustainment, and business/acquisition visions. Those visions were briefed to leadership and then turned into an executable plan, including a schedule with a critical path, documented requirements, architecture products, etc. Any proposed changes of the component systems to the plan must be assessed for the impact to the system of systems. The systems engineering process includes all the programs that are part of the system of systems and must be able to take into account technology insertions and changes in program direction.

Gaining Momentum

Another lesson learned is the value of a quick win to get momentum. The server consolidation effort is again a good example. In ABCS 6.4, as delivered to the first users, each system had its own server architecture. The servers were not integrated; there was a mix of unit- and PEO-provided equipment that was not necessarily compatible. PdM TBC took the lead to consolidate many of the servers into a single server solution, the Battle Command Common Server (BCCS). The Army Information Server, Maneuver Control System, Information

Dissemination Manager—Tactical, and Tactical Battlefield Enterprise Services were all servers in the ABCS architecture that provided information and enterprise services—such as e-mail—to the systems in the architecture. The servers cost about \$5.34 million; the new, consolidated BCCS came in at approximately \$3.47 million, a significant savings in hardware to the Army. The BCCS was integrated with ABCS, modular by design, and reduced the server and field support footprint. In order to execute the first step of server consolidation, products from four different offices were brought together under one office.

Need for Early Testing

The general philosophy for testing is to have as much testing as early in the development process as possible. Bugs found early in development cost less to fix and are much more likely to get fixed. A new approach for the system of systems was developed for subsequent ABCS versions. Prod-



ucts underwent stovepipe development test at contractor sites, then were sent forward for risk-reduction testing. A risk-reduction test is non-attributional testing of both functional and system-of-systems capability. It allowed program managers to test specific threads and new functionality and gave programs an opportunity to fix bugs and/or adjust techniques, tactics, and procedures. It also gave programs a chance to check that their services and clients interacted as expected and used the infrastructure as designed. Following the risk-reduction test, products moved into the formal test-fix-test and certification environment for attribution and reporting to leadership.

System-of-systems development and testing will increasingly be dependent on a consistent, coherent program objective memorandum estimate for the system of systems. In the future, funding will likely need to be allocated to the system of systems rather than to individual programs. Creating a system-of-systems program objective memorandum estimate was first done for the fiscal year 2008-2013 funding cycle by taking the migration plan, detailing out requirements for each product manager, prioritizing the work based on the system-of-systems efforts taking priority over stovepipe functionality, and then cross-leveling it so that gaps and duplications were identified. This resulted in an overall lower bill to the taxpayer, with interoperability among the systems built into the design. This would not have been possible without leader engagement, public design reviews, and a great deal of detailed systems engineering work done in advance.

The field support concept also underwent change as a result of lessons learned from fielding ABCS 6.4. The old field support process had individual system field support representatives embedded in units. Reducing the number of contractors in-theater was done by cross-training them so that they would be able to support more than one system. Support issues that could not be immediately resolved were inputted into a central system where they were worked by experts in the continental U.S.

Recommendations

Creating a vision to guide migration to the future is a key recommendation. The vision establishes a strategy and the process to be used to execute that strategy in advance of any hardware or software development. It should be a single, in-

tegrated technical, logistics, and business overarching vision created and agreed to by all major stakeholders, to include the organizations that fund the system of systems.

Systems engineering of the system of systems with the development of the associated architecture products must support the execution of the vision. Robust and integrated network services need to have detailed systems engineering that is focused on the architecture for net-centric warfare.

It is imperative for the system-of-systems management to continue to notify the Army leadership on testing, operational, funding, training, fielding and sustainment issues on a frequent basis. The Battle Command General Officer Steering Committee provided an excellent forum for ABCS. The forum was instrumental in getting Department of the Army-level guidance and issue resolution across agencies.

The transformation of ABCS from many stovepipe systems to a system of systems has not been just a technical issue; many different components needed to work together. There needed to be a network providing the commander with needed functionality, supported by trained soldiers, and with excellent technical support in the field, to keep it working under all conditions.

In developing and fielding a system of systems like ABCS, the acquisition community needs to remain aware of new technologies and best processes in the commercial

world. In addition, acquisition must adapt to the increasingly complex systems that warfighters use. A culture change is needed to make dealing with new technology and complex systems easier. A change in the thinking of all stakeholders in considering just a narrow system view to always considering the broader system-of-systems view is the first step. Stakeholders must always consider the impact of decisions on the next larger system of systems in all areas—development, training, fielding, testing, support, and funding.



The authors welcome comments and questions and can be contacted at harold.greene@us.army.mil and janet.greenberg@us.army.mil.